NUSC Reprint Report 6916
2 July 1990

Automation of Missile Flight Test Analysis and Report Generation

M. J. Desrosiers
Launcher and Missile Systems Department

H. A. Rosen
S. Lopes
Weapon Systems Technology and Assessment Department

Naval Underwater Systems Center
Newport, Rhode Island • New London, Connecticut

Reprint of a paper presented at the *AIAA/SFTR/DGLR/SETP Fifth Biannual Flight Test Conference,* Ontario, Canada, 22-24 May 1990.

AUTOMATION OF MISSILE FLIGHT TEST ANALYSIS AND REPORT GENERATION

Michael J. Desrosiers, Hillard A. Rosen, Steven Lopes
Naval Underwater Systems Center
Newport, Rhode Island

Abstract

To meet program objectives for consistent, timely, and thorough flight test reports during developmental testing of the Submunitions Tomahawk Land-Attack Missile, an Automated Flight Test Analysis and Report Writing Tool was developed by the Naval Underwater Systems Center, Newport, Rhode Island. This paper discusses the methods used to standardize flight test reports, design the flight test data base, select hardware and software components, reduce and store flight test data, filter missile telemetry (TM) data, standardize and automate analysis efforts, and generate the automated report. Results included: use of a relational data base to store large amounts of planned and actual flight test data; development of an autocorrelation routine to link TM data with planned mission data; application of heuristic techniques to filter individual TM channels; and standardization and automation of analysis efforts for submunitions impact-point density, boost phase, terrain contour matching, and terrain-following performance evaluation. Automated flight test report generation reduced the report preparation period from an average of 6 months to 2 months and improved reporting accuracy and consistency.

Introduction

The Tomahawk cruise missile is a subsonic missile capable of being launched from submarines, surface ships, aircraft, and land-based launchers against at-sea or land-based targets. Four Tomahawk missile variants have been established: the Tomahawk Anti-Ship Missile, designated U/RGM-109B; the Tomahawk Land-Attack Missile (TLAM) - Nuclear, designated U/RGM-109A; TLAM - Conventional, designated U/RGM-109C; and Submunitions TLAM, designated U/RGM-109D. The "U" prefix refers to a submarine-launched missile configuration while the "R" prefix refers to a ship-launched missile configuration.

U/RGM-109D Missile

The U/RGM-109D missile (hereafter referred to as the "109D" missile) carries combined effects bomblets (CEBs) rather than a single warhead to attack a land-based target. Developmental testing for the 109D missile was conducted between the fall of 1985 and the spring of 1988. The developmental testing (DT) program utilized an aggressive flight test schedule and required timely, detailed flight test reports.

Data Collection

Sample data collected for a Tomahawk flight test included: missile configuration, mission planning, launch platform, environment, and telemetry data; time, space, and position information (TSPI); and test launch problems and narrative information. Data items unique to 109D flight tests include CEB impact point data, which are used for terminal scoring analysis. The Tomahawk telemetry system transmits guidance and airframe analog data over high frequency radio waves to chase aircraft and ground-based receivers. The data recorded by multiple receivers are digitized and merged to produce a "best source" TM guidance and airframe data set.

The Tomahawk Data Base

A relational data base was established in 1982 to store many of the data items and analysis results collected for a Tomahawk flight test. This data base was modified during the 109D DT program to incorporate the special data collection and analysis efforts for the 109D missile.

Existing Analysis and Report Generation Techniques

The analysis tasks for a Tomahawk flight test were usually allocated to several supporting activities, both contractor and government, based on the flight test phase or test objective. In general, when several agencies shared the same analysis task, analysis methods and techniques varied greatly. When only one agency performed a given analysis task, analysis methods and techniques were not well documented and not well understood by other members of the analysis community.

Existing report generation techniques consisted of manually produced text, tables, and figures with the assistance of word processing and graphics specialists. Typographical errors were not uncommon because of the large amounts of data reported and the method of production. The nonstandardized report format allowed detailed discussions of unique flight test objectives and results; however, the variations and inconsistencies in report presentation style made it difficult for readers to quickly obtain desired information.

Standardizing the Flight Test Report

The first step in developing an automated flight test report writing tool is to define a standardized, or "boilerplate," report format. This can be done by reviewing existing, nonautomated flight test reports for the current weapon system (if available) or flight test reports from a similar weapon system. This review is used to identify

boilerplate text, tables, and figures that need to be presented consistently, along with flight test unique information, in the standardized report.

Identifying Boilerplate, Variable, and Drop-in Text

Flight test report text was identified as boilerplate, variable, or drop-in text. Boilerplate text remains fixed between flight test reports (e.g., generic weapon system descriptions). Variable text varies slightly between reports based on the flight test scenario used, specific test results, and other factors. Drop-in text is very specific to a given flight test, not likely to be repeated, and difficult to automate.

Developing Boilerplate Text. Boilerplate text is developed by using generic weapon system, test environment, and supporting system descriptions, and generic weapon system performance descriptions. Space can be reserved within the boilerplate text for inserting specific weapon system, test, and performance information if necessary.

Examples of boilerplate text in the 109D flight test report include: a generic missile variant description, "U/RGM-109D," which is used to cover both ship and submarine-launched flight tests; a standard list of contributing agencies and program objectives presented in the foreword section; and a standard design of the flight test performance evaluation criteria used in flight test analysis. Special text strings, "N/AP" and "N/AV," were inserted into boilerplate text when specific data items did not apply or were unavailable for a given flight test. This reduced the amount of boilerplate changes required for each report.

Developing Variable Text. To allow for variable text in the report, the factors that caused the variable text to change, such as the flight test scenario used, or the planned and actual flight test data, were first identified. The following questions were then asked:

1. Can the variable text be generated by software based on other data items stored for the flight test?

2. Could a string field or text table be used to store the variable text?

3. Should the final version of the flight test report be hand-edited?

Using a string field to store the variable text is useful for text that varies frequently between reports and is difficult to hand-edit into the final version of the flight test report. Hand-editing the report is a practical solution when the text is not likely to be repeated on subsequent tests.

An example of variable text in the 109D flight test report is the launch platform configuration description, which can vary according to data stored in the data base; this text was automated by software. Special fields and text tables were created to store variations in data source descriptions and flight test objectives. Information not likely to be repeated again in subsequent tests, such as the use of non-operational test equipment and adjustments made to terminal scoring results based on missile developmental test software (DTS) errors, were hand-edited into the final version of the report.

Identifying Drop-in Text. Drop-in text was used for detailed reporting of flight test anomalies and unique flight test objective results. In the 109D flight test report, section 4 was used as a stand-alone, free-format section with drop-in text, tables, and figures used to present detailed engineering results for the flight test. References to section 4 by other (automated) sections were hand-edited into the final report.

Identifying Standard Data Items, Tables, and Figures

Text, tables, and figures contained in existing reports were reviewed to determine what data sources are used, what data items are consistently available, and what text, tables, and figures are needed to support consistent evaluation of missile flight tests.

Identifying Data Sources Used. Data items presented in text, tables, and figures were linked to data sources, such as original data products, analysis results, and performance specification documents, to determine whether the data items were actual results, planned numbers, or specification values. Data items were separated according to the data source type for storage considerations.

Identifying Data Availability. Text, tables, and figures that present data items that are unique to a given flight test versus data items that are consistently collected were identified. Examples of inconsistently collected data items for the 109D flight test program were noseboom wind data and ground station wind data. Examples of consistently collected data items were launch platform, environment, mission planning, and telemetry data. The majority of data items, tables, and figures that were unique to a given flight test were presented in section 4 of the report.

Identifying Data Needed for Consistent Evaluations. Text, tables, and figures that present data items to support unique flight test evaluations versus standard flight test evaluations were identified. Examples of data items to support unique 109D flight test evaluations were wind data and aeronautical performance data to evaluate wind estimation and payload post-dispense missile controllability objectives. Examples of data items to support standard 109D flight test evaluations were terrain contour matching (TERCOM) data, digital scene matching area correlation (DSMAC) data, inertial and radar altitude data, and terminal accuracy results. Data items, tables, and figures to support unique flight test evaluations were presented in section 4 of the report.

Standardizing the Report Structure

The report was divided into standard sections, subsections, and annexes according to high-level versus detailed test results, generic weapon system descriptions versus actual test results, and standard versus unique flight test evaluations. Related information such as mission planning parameters, launch platform data, and in-flight test results were grouped together.

Report Format. Five standard sections were developed along with introduction and annex sections. The introduction discusses program objectives, and includes an executive summary of the flight test. Section 1 contains a summary of flight test objectives and missile performance. Section 2 contains an overview of Tomahawk weapon system anomalies and concerns. Section 3 presents standard flight test data items, tables, and plots related to system performance. Section 4 is a nonautomated section allowing a free-form presentation of detailed system and subsystem results. Section 5 contains an overview of test anomalies and concerns. The annexes contain reference material including missile and test configuration data, mission planning data, supporting weapon system descriptions, and evaluation criteria used for analysis.

Report Content. References between standard sections of the report are included in the boilerplate text. References to the nonautomated section of the report are hand-edited into the final version of the report. The standard report subsections cover all phases of the testing program. A data "not applicable" code ("N/AP") is used when an individual subsection does not apply to a given flight test. An example of this for the 109D flight test program is the recovery phase for a live submunitions shot. This feature improves the automated report writing capability by reducing the number of hand-edits required to the final flight test report.

Data Base Design and System Architecture

There are many decisions to make when designing a data base and choosing a hardware and software system to automate flight test analysis and report generation. These decisions can affect data entry, system performance, security procedures, and report quality.

Data Base Structure

The factors considered when designing the data base oi 109D flight test data items included: selection of a relational data base system; online access to flight test data; grouping related data items; the ability to link planned data with actual results; and software performance.

Relational Data Base Structure. Each 109D flight test has the potential for a varying number of navigation update points, targets, test objectives, anomalies, and missile and test configuration data items. There also can be varying length test and problem narratives. A relational data base structure was better able to track this information than a nonrelational data base structure.

The majority of 109D flight test data items was stored in the Tomahawk data base, which had already been developed. This data base currently consists of 1005 data items in 113 tables, storing such things as planned mission data, launch platform data, environmental data, in-flight analysis results, flight test problems, and narrative information. Each table uses one or more primary keys -- fields needed to uniquely identify a single record of data. Some tables also use one or more secondary keys -- fields that are part of a primary key in another table. Both primary and secondary keys are used to link information between data base tables.

Online Access to Flight Test Data. New data tables were designed to store 109D flight test range data, including guidance TM, DSMAC TM, and TSPI data. A relational structure similar to that of the Tomahawk data base was employed. Additional data tables were also designed to store 109D flight test report-specific data items. These new tables were frequently referred to as "offline," or "unofficial," data base tables to distinguish them from the approved set of Tomahawk data base tables; each table, however, was accessible by flight test analysis and report writing software at all times.

A large storage device was required to achieve this. The Tomahawk data base holds approximately 4 megabytes (MB) of flight test data. Range data for a single flight test requires about 5 MB of storage space. Because of the data reduction and filtering processes of the Tomahawk automated flight test report writing tool (RWT), the storage requirement for range data doubles (up to 10 MB). RWT software itself is about 2 MB; another 2 MB of available disk space is required when running RWT for temporary data file storage. In all, at least 20 MB of online storage space is required.

Grouping Related Data Items. When designing new data base tables, related data items were grouped by considering the following factors: data source; data sampling rate; data entry method; report use; and classification.

Grouping data items by data source was done for all Tomahawk data base data and offline tables. Grouping data items by data source kept related data items together and made data entry easier.

Grouping data items according to sampling rate reduced data redundancy and improved data access time. For example, radar altitude data are recorded at 8 samples per second (SPS), while inertial altitude is recorded at 2 SPS. Combining these data items in the same table would require storing redundant inertial altitude values, or compressing multiple radar altitude values into each record of inertial altitude data.

Grouping data items according to data entry method improved the data entry process. For 109D, the tables with large amounts of flight test data were filled by automated entry methods, while tables with small amounts of flight test data were usually filled by manual methods. This allowed data entry for several data tables to be done in parallel.

Grouping data items by report use allowed data items used specifically for the 109D flight test to be separated from data items that are also useful for trend analysis.

Grouping data items by classification may be a security requirement of the agency processing the flight test data. For example, Chief of Naval Operations Instruction 5239.1A recommends that all UNCLASSIFIED and CONFIDENTIAL information be stored on separate write-protected media when processing SECRET data on an Automatic Data Processing (ADP) system operating in the "dedicated" mode.[1] When processing 109D flight test data, the entire Tomahawk data base is treated as SECRET and all output is manually reviewed and reclassified as appropriate for use in the flight test report; this procedure is called a "system high" mode of operation. Flight test data were also separated by classification level into different tables or directories in order to make data backups and classification of outputs easier.

Linking Planned Data with Actual Results. The evaluation of terrain-following, TERCOM and DSMAC performance, and terminal accuracy for the 109D flight test required the ability to compare planned data with actual results. Because related data items were grouped, it was necessary to link data items from in-flight test results and range data tables with similar items from planned mission data tables. This was done by using similar key structures in these tables. For example, guidance telemetry data were correlated to planned mission profile data by using trajectory segment number keys.

Examining Software Performance. The trade-offs between reducing data redundancy and improving software performance were considered when designing data base tables.

Reducing data redundancy was desired to improve the data entry effort and reduce the number of data changes needed when the source data changed.

Reducing the search time to obtain data was desired to improve software performance. This was initially done by relaxing the data redundancy requirements in order to store planned information with range data results; this reduced the number of tables that had to be accessed when comparing planned data with actual test results. A more desired method to reduce search time, however, was to improve record linking techniques between tables. For example, a table summarizing the actual trajectory segment information was created; this table contained the record number for the start of each actual trajectory segment in the guidance telemetry table.

Hardware Considerations

The major factors considered in choosing a hardware system to perform automated analysis and report generation were: data storage requirements; security requirements; printer and plotter requirements; and transportability of data between data providers and data receivers.

Data Storage Requirements. To have online access to the complete set of Tomahawk data base tables and flight test range data during automated analysis and report generation, approximately 20 MB of storage space was required. For most systems this meant that RWT software could be used only on one flight test at a time. Flight test range data, requiring up to half of this storage requirement, were archived following completion of the flight test report.

Security Requirements. When processing classified material, security requirements of the processing agency may require the use of an ADP system in a secure environment, the removal of all classified data from the ADP system for proper storage, and declassification of the ADP system. Monitoring of the ADP system during classified processing may also be required.

Similar procedures followed by the Naval Underwater Systems Center (NUSC) required the use of a removable hard disk or removable cartridge device for storing 109D flight test data. All software routines were required to run within an 8-hour period for system monitoring. For these reasons, an IBM PC-AT with an 80286 processor and 30-MB removable hard disk was chosen in 1986 for the RWT system. In 1989, IOMEGA Bernoulli Box II devices with 20-MB removable disk cartridges were added to the system for increased data storage capacity and data transportability.

Printer and Plotter Requirements. High resolution plots (up to 300 dots per inch) were required for Terrain-Following Profile, Cruise Phase Detail, and Bomblet Impact Points/Density Analysis plots. Near letter-quality printouts and the condensed landscape capability for large report tables were also required. Finally, the ability to integrate text, tables, and plots into a final report was considered.

Initially, a Houston Instruments Model DMP-29 plotter was chosen in 1986 for producing the plots. This was replaced by the Hewlett Packard LaserJet Series II printer with 2 MB of memory in 1987. The LaserJet met both the plotter and printer requirements and was fairly easy to use for integrating text, tables, and plots.

Data Transportability. The ability to read 9-track, 1/2-inch, 1600 or 6250 bits per inch, range data tapes for TM or TSPI data was required. For this, an external tape drive was connected to the PC system.

The ability to transfer flight test results between analysts and report writers was also desired. This was usually done on high-density and low-density diskettes. When Bernoulli drives became popular among 109D flight test analysts, complete sets of flight test data could be transferred on a removable disk cartridge.

Software Considerations

The major factors considered when choosing software to perform automated analysis and report generation were: report generation capability; compatibility with existing data base and analysis software; the ability to handle large data files; simultaneous access to a large number of data tables; interactive graphics; and random access memory (RAM) requirements.

Report Generation Capability. Some of the report generation features considered were: the ability to merge data base data items with boilerplate text; the ability to make boilerplate format changes; and the ability to merge text, tables, and plots.

Software Compatibility. Compatibility with existing Tomahawk data base application software and the interface capability with other languages, such as C, FORTRAN, and PASCAL, were considered.

KnowledgeMan (KMAN) data base software by Micro Data Base Systems Inc. was chosen in 1984 to support the Tomahawk data base. Existing data reduction and analysis software routines were written in FORTRAN and PASCAL. KMAN has interface capability with Lattice and Microsoft C.

Ability to Handle Large Data Files. Direct access to large data files was required to reduce, filter, and analyze 109D flight test data. Up to 50,000 records of data could be required for some telemetry data items.

Ability to Handle a Large Number of Tables. Simultaneous access to a large number of tables was required in order to combine planned data with actual results. KMAN currently allows up to 45 tables to be open at the same time.[2]

Interactive Graphics. The capability to display data graphically while running automated analysis software and to allow the analyst to modify data points in the table based on data points selected from the screen was desired. KMAN could display data graphically while running analysis software, but the user could not interact with screen graphics to modify data points.

Detailed plots for the final 109D report were done using KMAN to read the data base tables and produce American standard code for information interchange (ASCII) data files. Software to read the data files and produce plots was written in Lattice C using graphics device drivers available from Graphics Software Systems (GSS).

RAM Requirements. Conservation of RAM when running automated analysis and report generation software was necessary due to the 640-kilobyte addressable memory limitation of disk operating system (DOS) software. KMAN's RAM requirement allowed several functions, written in Lattice C, to be loaded into memory. Plotting routines could not be run in the same session as data reduction and analysis routines, however, because of the additional memory requirements of the graphics device drivers.

Data Reduction and Storage Techniques

After designing the data base structure and identifying the hardware and software system needed to automate analysis and generate the flight test report, several data reduction and storage techniques were considered. These included: automated data entry techniques; techniques for automated correlation of planned data and actual results; manual data entry procedures; and data management techniques.

Automated Data Entry

Because of the large amount of data collected for a 109D flight test, including both planned data and actual results, automated data entry was a requirement. Several techniques for automated data entry were considered. These included examining the use of alternative systems to reduce data; developing methods for appending data to the data base; filtering gross noise; and developing methods to handle missing or excessive samples of data.

Alternative Data Reduction Systems. Alternative hardware and software systems with larger storage space and faster input/output (I/O) to reduce and reformat the original data sources were initially used because of the vast amounts of range data collected and the computational intensity of data reduction.

Original telemetry data can require up to 30 MB of storage space. Because of this, a Digital Equipment Corporation VAX 11/780 was initially used to reduce and reformat the 109D telemetry data. ASCII text files were created and transferred by magnetic tape to the PC for uploading to the data base. Some of the disadvantages of this method were the time and logistics required to transfer the ASCII text files to the PC system.

Appending Data to the Data Base. There were two possible methods of appending data to the data base: using ASCII text files with the KMAN ATTACH command, or using an interface language (such as Microsoft C) to read the original data files, build the required KMAN tables, and directly append data to the data base.

Initially, FORTRAN code was developed to read the 109D telemetry data and build ASCII text files. Besides being time consuming, there were problems with noise values overflowing the FORTRAN format statements. When this happens, asterisks appear in the data file, and KMAN will not accept the asterisks if an integer or real

number is expected. Therefore, data noise had to be filtered to prevent these overflows -- either by hand-editing the output text files, or by adding FORTRAN code to test for noise. In either case, the flag value assigned to noisy data could not exceed the original FORTRAN picture size in order to maintain the fixed record length needed for tape I/O procedures. This resulted in extra automated data analysis code on the PC system to distinguish reasonable data values from various length flag values.

Subsequently, Microsoft C code was developed that interfaces with KMAN software and directly converts original 109D TM data into data base records. This method was faster because data transfer logistics and lengthy attaches of text files were eliminated. This method also allowed the original TM data to be stored "as is" -- without having to use special flag values to replace noisy data.

<u>Gross Noise Filtering.</u> Gross noise filtering was accomplished by comparing actual flight test data with fixed upper and lower limits. Data values exceeding these limits were replaced by a standard flag value recognizable by filter, analysis, and report writing software. Things to consider when implementing gross noise filtering are when to apply gross noise filters and determining the fixed limits needed.

Gross noise filtering can be implemented in the automated data entry software used to build the flight test data tables, or in subsequent software used to copy the flight test data tables prior to applying more sophisticated filter software. The latter method was preferred because the original flight test data values were preserved and would not have to be regenerated if the gross filter limits required modifications. Preserving the original data values was also useful for performing comparisons with filtered data.

The gross filtering limits were determined from applicable 109D missile performance specifications, or from range restrictions applied to 109D flight tests. Noise values were usually replaced with a "null" value -- a flag value consisting of 14 nines (the largest value that can be stored in a KMAN number field).

<u>Handling Missing or Excessive Data Samples.</u> Data noise and the process of merging multiple telemetry data sources frequently resulted in missing or excessive sampling of data values. These fluctuations were corrected by the automated data reduction software for the following reasons:

1. the seek time for a specific data item by subsequent analysis and plotting software can be significantly improved; and

2. the performance of subsequent data filtering software can be significantly improved by using fixed limits on changes between consecutive data samples.

These enhancements are possible only if a fixed data sample-to-record number relationship applies.

For example, if a data value is telemetered at 2 SPS and stored at 2 records per second then a request for the data value at 1 minute following the present position requires a forward move of exactly 120 records in the data table. Also, the change between consecutive data values may be interpreted as a rate for certain parameters -- such as vertical velocity for inertial altitude data. These rates of change between consecutive data parameters may have associated limits according to a performance specification document or a reasonable limit. If the relationship between the data sampling rate and record number is consistently maintained, data noise can be identified when changes between consecutive data values exceed the fixed rate of change associated with that parameter.

The data sampling rate-to-record relationship for 109D flight test data was consistently maintained as follows:

1. a 4-to-1 ratio of radar values (8 SPS) to inertial altitude values (2 SPS) was maintained during the automated data entry of 109D telemetry data; and

2. a "Greenwich mean time (GMT) adjustment utility" was performed by data reduction software to add blank (null-filled) records during data drop-out periods and eliminate excessive records during periods where multiple raw TM tapes had been merged.

These adjustments were made to "working" data tables -- copies of the original TM data tables.

Autocorrelation of Planned and Actual Data

Following the automated entry of flight test data, application of gross filter limits, and adjustments for data sampling rates, an automated correlation routine was applied to identify and store the linking information between 109D guidance telemetry data and planned mission trajectory segment data. The steps involved in developing this correlation technique were (1) examining potential methods to compare planned data and actual results; (2) correlating planned and actual trajectory segments; and (3) allowing the analyst to review and modify the correlation results. The results of this autocorrelation routine formed the basis for subsequent comparison between planned data and actual results and for evaluation of the missile in-flight performance.

<u>Potential Methods to Correlate Planned and Actual Data.</u> Some potential methods identified to compare planned and actual flight test data were:

1. comparison of planned and actual flight test modes and phases, such as boost, cruise, terminal, and recovery phases;

2. comparison of planned waypoints with actual (missile telemetered) latitudes and longitudes;

3. comparison of planned trajectory segment* numbers and commanded clearance pointers; and

4. comparison of planned and actual vertical profile data.

The first two methods identify general areas of transition in the actual flight test data that can be compared with planned performance information; however, these areas of transition were not specific enough for 109D flight test analysis.

The fourth method identifies very specific areas of transition in the actual flight test data but requires the storage of detailed planned vertical profile data that were not available for 109D flight tests; also, this method cannot be used when the missile flies offcourse.

The third method was chosen to identify transition points between actual trajectory segment boundaries by using changes in the telemetered commanded clearance pointer (J984U) values. Actual trajectory segment information was compared to planned trajectory segment information stored in the data base for flight test evaluations.

Correlating Planned and Actual Trajectory Segments. The delta times between planned time of arrival (TOA) values for trajectory segments were compared with delta times between transitions in the J984U values. An array containing delta times for up to 100 transitions in J984U values was compared with the planned TOA deltas by using a linear correlation model to find the "best match." The best match is the one with the highest correlation value; any value between 0.75 and 1.00 represents a strong correlation.[3]

Following correlation, noisy transition points in the J984U values are resolved using planned TOA values.

Analyst Review of Correlation Results. Prior to data correlation, the software prompts the user to confirm the critical data items needed, such as the number of operator-entered waypoints used in the flight test. During autocorrelation, the user can monitor the correlation values computed by the algorithm. Finally, the results of automated correlation between planned trajectory segments and actual commanded clearance pointers are stored in a data base table for the analyst's review.

Manual Data Entry

Manual data entry techniques specific to the flight test report writing task were documented in the RWT Analyst and User Guide developed by NUSC. Special guidelines were given in the following cases: when the data item was not part of the official Tomahawk data base; when the required data did not match the existing Tomahawk data base definition; or if the data entry technique did not match the existing conventions of the Tomahawk data base.[4]

A specific data entry technique was developed to allow unfilled data items to be categorized as not applicable, not available, undetermined, or pending. This allowed the appropriate data status code ("N/AP,""N/AV,""UND," or "PND") to be inserted into report text, tables, and plots.

Data Management Techniques

Data management techniques were applied to identify data entry requirements, prioritize the data entry efforts, and track data modifications.

First, data base fields were linked to the input requirements of data reduction, analysis, and report generation software. Data entry tasks were then prioritized so that time-consuming software routines could be run as soon as all data input requirements were met. Finally, a log of the data entry process was kept to identify data table completeness and data modifications; this log was used to determine what analysis or report generation software needed to be redone because of data changes.

Data Filtering

Data filtering was required to reduce telemetry data noise points prior to the application of automated flight test analysis and report generation software. This was done by identifying the causes of telemetry data noise, examining the characteristics of data noise, identifying the potential filtering methods, and defining filter philosophies and audit trails. Examples of the filtering methods developed for 109D inertial and radar altitude data are discussed in this section.

Causes of Telemetry Noise

Telemetry noise comes from several sources. One is the missile hardware devices, along with the method of telemetry data sampling and buffering internal to the missile. For example, radar altitude data can be noisy due to bad radar altitude feedback that is not eliminated during the Kalman filtering process.

A second source is the introduction of noise during the TM transmission process. This could be related to the location of the TM receiving stations relative to the missile position, "multi-pathing" of the telemetry signal, or other unknown environmental effects. Multi-pathing of the TM signal occurs when the received TM signal is reflected from other surfaces and received out of phase with the original signal.[5] Poor location of the receiving stations relative to the missile position results in weaker TM signal strength and increased TM signal multi-pathing.

The methods of processing the recorded telemetry data also affected data quality. For example, excessive data samples were introduced when TM data from different

*A trajectory segment is a portion of the missile flight path defined by a specific set of flight parameters.

receiving stations were merged to produce a "best source" TM data tape.

Finally, the rate of data transmission affected the quality of telemetry data. During the 109D DT Phase, the rate of TM data transmission was increased from 44 to 384 kilobits per second. This increased the difficulty of ground station processing and reduced the quality of the recorded data.

Characteristics of Telemetry Noise

The behavior of data for individual 109D telemetry parameters was examined to identify data noise points. Obvious noise data points were observed as one of the following:

1. single point data spikes in the TM channel;

2. continuous, steady-state values that may exceed the minimum or maximum limits established for the data parameter; and

3. rates of change between data samples that exceed missile performance capabilities or defy laws of physics.

Questionable data noise points were also found -- data values that were within the range of acceptable limits, and within acceptable rates of changes between adjacent data points, that were potential noise points. A trained analyst was required to identify those points and additional analysis was needed to determine data validity; frequently, a final determination could not be made.

Good data points, on the other hand, were usually characterized by slowly changing values, or trends, between data points. Exceptions to this were TM channels for missile pitch and roll; these varied too rapidly to distinguish good data values from noise points. Automated data filtering was not attempted for these parameters.

Methods of Filtering Telemetry Noise

Several methods of filtering telemetry data were identified. These included using secondary telemetry channels for comparison; using planned mission data; and using heuristic techniques.

Using Secondary Telemetry Channels. This method uses secondary TM channels, related to analysis, for reducing data noise. For example, the vertical velocity channel could be used to filter inertial altitude data since vertical velocity is the rate of change in inertial altitude. Similarly, north and east velocity channels could be used to filter latitude and longitude data. The problem with this method, however, is that data noise can appear simultaneously in several channels, or sporadically in isolated channels depending on the cause of data noise; this makes the results of comparisons with other data channels unreliable.

Using Planned Mission Data. This method compares telemetered values with planned mission data for data filtering. For example, planned waypoint data could be used to filter telemetered missile latitude and longitude position. Planned Mach numbers and vertical velocity limits could also be used to filter telemetered Mach and inertial altitude values. However, problems occur with this method when the missile does not follow the planned flight path (because of manual override commands or anomalous missile behavior), when environmental effects alter planned missile performance, and when actual missile performance does not compare well with planned performance parameters. In some cases, missile performance results outside of the planned performance envelope were of more interest to flight test analysts and program managers than other results, making this method of data filtering undesirable.

Using Heuristic Techniques. This method uses the characteristics of individual telemetry parameters to derive limits for data values and rates of change between data values. The acceptable limits and rates of change between data values can be determined from missile performance specification documents, general laws of physics, restrictions imposed by the test environment, or by establishing "reasonable" limits and rates of change (based on comparison studies between good and bad sets of data samples). These limits and rates of change are used to distinguish "good" data sets from areas with potential noise points. In some cases, a mathematical model can be used to describe the behavior of data between good data sets; this model could also be used to identify obvious noise data points.

The disadvantages of this method include the accuracy of the limits and rates of change used to identify good data sets, and the accuracy of mathematical models used to predict the behavior of data between good data sets. For example, performance specifications may not accurately reflect actual missile performance, and reasonable limits may change according to the test environment or other factors. Some of these problems can be overcome by combining heuristic techniques with other filtering techniques, and by using lookup tables with filter limits that can be modified by the analyst.

Defining Filter Philosophies and Audit Trails

The philosophies for handling data noise and tracking the results of data filtering must be determined.

Filter Philosophies. The philosophies of data filtering include: whether to change questionable data values or just obvious noise data points, and whether to replace selected noise data points with a flag value or an interpolated result.

For the 109D TM filter software, "questionable" data values -- values that appeared to be noise data points but that were within the performance capability of the 109D missile -- were left unchanged. This was accomplished by expanding the range of acceptable data values and rates of

change between data values used to filter data. Questionable data values were considered significant to the performance analysis of the 109D missile and required further examination by an analyst. In some cases, questionable data values were written to a log file for an analyst to review.

For the 109D TM filter software, noise data values were usually replaced with null values. In general, interpolated results were not trusted because a mathematical model could not accurately predict the behavior of data over a noisy area. One exception to this was the inertial altitude filter; this filter uses linear interpolation over short noise segments in the cruise phase of flight, and a second degree curve fit to model data in the boost phase and terminal areas of flight.

Audit Trails. Audit trails for 109D filtering results were accomplished either by archiving copies of the original (unfiltered) data files or by maintaining filter-specific log files that contained a record of all data changes made, and all questionable values left unchanged. Archived copies of original data files could be used for producing comparative data plots of before and after filtering results, and for refiltering data using modified filter limits or philosophies.

Results and Examples

The automated approach was to identify the telemetry noise data points "just as an analyst would by eye." To do this, heuristic techniques combined with comparisons between TM data channels and planned mission data were used. Performance specifications and reasonable limits were used to identify an initial set of good data values. The filter would then slide forward until a potential noise point was found. At that point the algorithm searched forward until another set of good data values was found. Then the algorithm reexamined the potential noise data points and, in most cases, used high and low limits around a linear transition between the good data sets to accept or reject data values. Specific examples are given for the inertial and radar altitude filters.

Inertial Altitude Filter. Inertial altitude is the missile height above sea level computed by the cruise missile guidance set (CMGS). The CMGS has the ability to update the inertial altitude following TERCOM mapsets, DSMAC scenes, and vertical-only update points. Valid 109D inertial altitude values appear in the telemetry data as slowly changing data values except when vertical updates are applied. The rate of change between data points can be interpreted as the missile vertical velocity, which is limited by missile performance capabilities. Mission planning parameters, wind contributions, and inertial altimeter accuracy also affected vertical velocity measurements. Mission planning parameters applied mainly to the cruise phase of flight. Vertical velocities for the boost and terminal areas were controlled by the missile DTS.

Limits on the rates of change between consecutive data values were used to identify good sets of inertial altitude data. These limits were established based on missile performance specification with a tolerance for other contributing factors. Mission-planned vertical velocity limits were not used to filter inertial altitude data because this would have eliminated data points significant to performance evaluation of the 109D missile. Also, different limits for maximum and minimum allowable inertial altitude and vertical velocity values were used for the boost phase and terminal areas of flight.

Models to predict the behavior of data over noisy areas were established; these were also used to identify obvious noise data points. For the boost phase and terminal areas of flight, a second degree curve fit was applied; for the cruise phase, linear interpolation was used. For both models, limits were set on the maximum number of points for which interpolation could be performed. In the cruise phase, this varied by the terrain-following mode of the missile. Consecutive noise data points that exceeded the limit for the given flight test mode were changed to null values.

Special filtering methods were required to handle areas where vertical updates were applied. For this, planned mission data were used to determine the approximate location of the vertical update. Telemetered vertical update values were used to determine the specific location of the vertical update and the magnitude of the vertical update applied to the inertial altitude channel. With this information, the previous filtering method was modified so that the vertical update point was preserved.

Figure 1 is a sample plot of unfiltered inertial altitude data in the cruise phase and terminal area of flight.

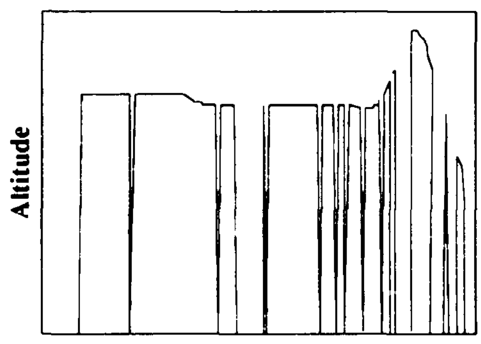

Fig. 1. Unfiltered Inertial Altitude Data

Figure 2 is a sample plot of the data from figure 1 after filtering. The data gap in the middle of the plot was a noisy area for which data interpolation could not be performed.

Radar Altitude Filter. Missile radar altitude is the missile height above ground level measured by the missile radar altimeter. Radar altitude data are sampled at 8 SPS versus 2 SPS for inertial altitude data. Radar altitude

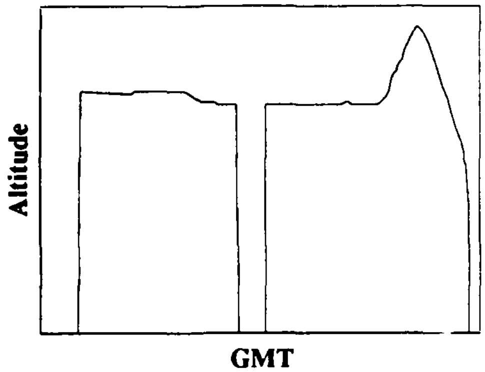

Fig. 2. Filtered Inertial Altitude

data vary more rapidly than inertial altitude data because of the change in terrain elevation and missile inertial altitude. Also, 109D radar altitude data were noisier than inertial altitude data because of the missile radar altimeter system.

A fixed limit for the rate of change between consecutive radar altitude samples was used to identify good sets of radar altitude data. This limit was established based on a reasonable terrain slope (Θ), and the maximum rate of change in inertial altitude. The limit was computed as follows:

1. assume the missile is diving at a maximum rate;

2. assume the missile is travelling in the horizontal direction at a maximum rate;

3. assume the missile is flying over a land mass with increasing slope Θ; then,

4. the maximum change in radar altitude is the sum of the missile altitude and the terrain elevation change in one-eighth of a second.

Figure 3 shows the components involved in this calculation.

A variable limit for the rate of change between radar samples, based on a percent of the average radar value collected, was needed for areas of lower terrain elevation, such as overwater and coastal areas of flight, in order to reduce the level of data noise accepted.

Although these fixed and variable limits for the rate of change between radar samples were used to identify good sets of radar data, larger rates of change (and therefore greater slopes of terrain) could be accepted between good sets of radar data.

Because of the unpredictability of terrain elevation changes, no standard mathematical model could be applied to describe the behavior of radar altitude data over a noisy area. Rather, various heuristic techniques were applied to filter data points in a noisy area depending on the size of the noisy area.

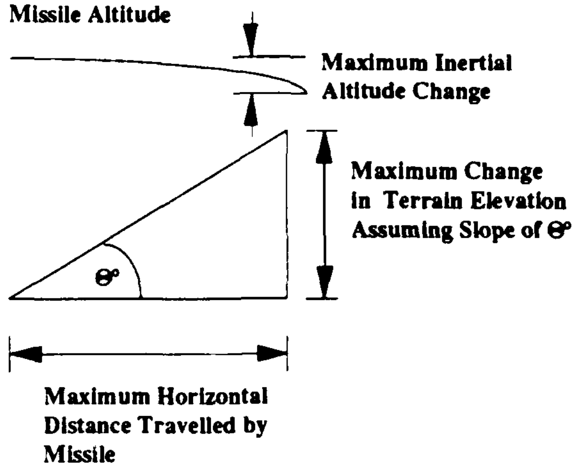

Fig. 3. Components Used to Calculate Maximum Change Between Radar Altitude Samples

For a single potential noise point surrounded by good radar data sets, linear interpolation was used to overwrite the questionable value.

For two potential noise points surrounded by good radar data sets, the average of each good set was computed. If either value in question lay outside of the upper and lower limits formed by the computed averages, that data point was considered a noise data point.

For three or more potential noise points surrounded by good radar data sets, the following algorithm was applied:

1. the standard deviation and average for each of the two good radar data sets were computed;

2. five times the larger standard deviation, or the number 25, whichever is greater, was used as an offset value;

3. the offset value was added to the higher average radar value and subtracted from the lower average radar value to create an acceptance window;

4. potential noise points within the acceptance window were accepted;

5. if two consecutive values were accepted, and the rate of change between them was less than the maximum rate of change (fixed or variable), the average of those two points was used to compute a new acceptance window; and

6. the process continued until all data points within the noisy area were examined.

The "#MARK" logic field contained in each radar altitude record was used to identify noise data points; this field was set to "TRUE" by the filter software. Records labelled TRUE were ignored by subsequent analysis and plot software.

More than 99 percent of the radar altitude noise data points were eliminated. Additional noise data points or corrections to the filter results could be made by modifying the #MARK field. Except for the single noise data point case, the original radar values remained in the existing data table. In any case, original values remained in a separate table for comparative purposes or for refiltering using modified filter limits and philosophies.

Standardization and Automation of Analysis Efforts

Analysis efforts required for a 109D flight test report were standardized and automated whenever possible and practical in order to reduce the report preparation time. This was done by reviewing existing analysis efforts, identifying candidate analysis tasks for automation, standardizing and automating the analysis efforts, and storing analysis outputs for an analyst's review and report use. Additional benefits of standardized and automated analysis efforts included more consistent analysis results and generation of additional information not available through manual techniques.

Reviewing Existing Analysis Efforts

Existing, manually produced flight test reports for other Tomahawk missile variants were reviewed, and Tomahawk flight test analysts were interviewed, to determine what analysis tasks were being performed and what computer algorithms and documentation existed.

Existing Analysis Tasks. The following analysis tasks were performed by one or more activities within the Tomahawk community to support flight test reporting:

1. TERCOM performance evaluation, which determines the adequacy and accuracy of the navigation update information passed to the CMGS following a TERCOM mapset;

2. DSMAC performance evaluation, which determines the adequacy and accuracy of the navigation update information passed to the CMGS following a DSMAC scene;

3. terrain-following performance evaluation, which determines the missile's ability to maintain altitude control over various types of terrain;

4. launch phase events analysis, which evaluates the sequence of telemetered missile events during the boost phase of flight;

5. aerodynamic performance evaluation of the missile under various flight conditions; and

6. terminal area performance analysis, which evaluates the missile's ability to attack a predefined target.

Existing Algorithms and Documentation. Existing computer algorithms typically consisted of small programs with manual data inputs used to perform a computation related to a specific analysis task. One exception to this was a semiautomated terrain-following performance evaluation program, installed on a mainframe computer, that had direct access to unfiltered Tomahawk telemetry data. Since the program did not have access to planned mission information, significant user interaction was required to identify valid 10,000-foot data segments; also, the program did not perform well in areas with significant telemetry noise.

Documentation for most analysis efforts did not exist. The analysis techniques involved were usually handed down, without documentation, from experienced analysts who might have left the project. Exceptions to this were the TLAM terminal analysis scoring plans, which were co-authored by several agencies in the Tomahawk analysis community. The program performance specifications and program design specifications for the missile DTS could be used to understand how the missile was supposed to work; however, these documents were often out of date during the 109D DT Phase.

Identifying Analysis Tasks for Automation

Existing analysis efforts were reviewed to determine whether they were possible and practical to automate. Several characteristics were examined for this determination.

Evaluating Task Characteristics. Analysis tasks that are performed infrequently for flight test reporting were not considered practical for automation; existing analysis techniques could be used to generate drop-in report results if necessary. Analysis tasks that require very repetitive computations and a large sample of data points are good candidates for automation. The degree to which automation was possible depended on the sensitivity of the analysis results to TM data noise and data interpretation. For example, an analysis algorithm that required a single value from a TM channel containing noisy, nonrepeating data was harder to automate than one that used values from TM channels containing less noisy, repetitious data. Similarly, analysis algorithms that required adjustments to the input data or calculated results, based on the specific missile hardware or software used or other factors, were more difficult to automate.

Results. Each of the existing analysis tasks listed previously applied to the evaluation of the 109D missile. For these tasks the following characteristics were identified:

1. TERCOM performance evaluation was consistently performed, repetitive, and used a large sample of data; results were insensitive to low amounts of TM noise; data input and analysis results did not require subjective interpretation;

2. DSMAC performance evaluation was consistently performed (for 109D missiles), repetitive, and used a large sample of data; results were sensitive to TM noise in a few channels; most data input and analysis results were not subject to interpretation;

3. terrain-following performance evaluation was consistently performed, very repetitive, and used almost the entire set of inertial and radar altitude data; results were not sensitive to low amounts of TM noise; subjective interpretation of data input and analysis results was not required;

4. launch phase events analysis was consistently performed, not very repetitive, and used a small sample of data; results were very sensitive to data noise; some subjective interpretation of the data input values was required;

5. aerodynamic performance evaluation was inconsistently performed, nonrepetitious, and used varying amounts of TM data; results were sensitive to TM noise and data interpretation; and

6. terminal area performance analysis was consistently performed, repetitive for the number of targets involved, and used a small sample of TM data; results were very sensitive to TM noise; considerable interpretation of results was needed because of such things as submunitions dispensing anomalies, impact-point measuring techniques, and missile DTS glitches.

For the 109D automated flight test analysis and report writing tool, TERCOM and terrain-following performance evaluation were fully automated. Launch phase events analysis consisted of automated extraction of TM events only. Automated DSMAC and terminal area performance analysis are still under development.

Standardizing the Analysis Efforts

Analysis efforts were standardized prior to automation. Standardization consisted of identifying the logic needed to perform the analysis task, resolving differences in data interpretation and analysis techniques between analysis activities, and identifying valid data sets to use.

Identifying Analysis Logic. Analysis logic consisted of the decision-making tools needed to interpret the data collected for a flight test, compute additional data items, and validate the analysis results. Examples of the decision-making tools required were:

1. knowledge of the relationship between planned data and actual results;

2. knowledge of the relationship between actual results and the particular flight test scenario, missile, and test support equipment used;

3. knowledge of the critical events needed to perform the analysis task;

4. knowledge of the expected sequence of events for the given analysis task;

5. awareness of related missile performance anomalies that could affect the analysis task; and

6. knowledge of the success and failure criteria used for missile performance evaluation.

Not all of the 109D decision-making tools required for a given analysis task could be automated. However, information critical to the decision-making process of analysis software was stored in a separate log table for an analyst's review, or the results of automated analysis software could be interactively overwritten by an analyst having a better understanding of the flight test.

Resolving Differences in Data Interpretation and Analysis Techniques. Differences in data interpretation can arise when different data sources are used. For the 109D flight tests, telemetered data from different sources and different iterations of planned mission or survey data were used, and this significantly altered terminal area performance evaluation. Some analysts had access to little-known, key information concerning abnormalities in the TM data that affected data interpretation. Other analysts had access to undocumented missile DTS changes that could affect missile performance. Most of these problems, once identified, were corrected by making changes to data distribution plans.

Analysts tended to use different mathematical techniques for analysis. One example was the different geodetic methods for computing the distance between two latitudes and longitudes -- great circle, rhumbline, and others. Data interpolation techniques also varied -- linear, curve-fit, and others. These differences had to be resolved to standardize the analysis effort.

Identifying Valid Data Sets. Once analysis logic has been defined, and differences in data interpretation and analysis techniques are resolved, it is necessary to establish criteria for defining valid sets of data points to analyze. For the 109D, this involved identifying and rejecting data noise points, correcting data for flight test-specific parameters, and evaluating specific data validity criteria for the given analysis task.

Data noise points were identified by methods similar to those used by data filtering software -- by using reasonable limits or missile performance specifications to test the validity of individual data points and groups of data points. Groups of data with excessive noise points (determined interactively by the user or by software) were rejected, and the search for additional data continued.

Certain data parameters were sensitive to TM timing problems or changes in the test scenario, test

environment, or missile software used. Such parameters needed to be identified by software, and verified or corrected before processing continued.

Each analysis task may have specific data set validity criteria. These limits may be used to quickly reject candidate data sets before further analysis is done.

Automating the Analysis Efforts

Automated analysis software was developed following standardization of the analysis efforts. Specific data validity limits and top-down programming were used to produce efficient and easily modifiable software. Allowing for analyst interaction improved the accuracy of automated analysis results.

Analysis Software Design. Top-down programming is the process of stepwise refinement of an algorithm into successively smaller pieces until software can easily be written.[6] Top-down programming using a logic flow similar to the standardized analysis procedures was used. The software was simplified as much as possible so that future modifications to software logic would be easier to make. Constant coefficients and data validity limits were stored in global variables; key logic decisions made by the software were placed at high levels. Analyst input or lookup tables were used in some cases to store analysis parameters and filter limits that changed frequently. Frequently performed software modules that were computation intensive, such as the terrain-following autocorrelation routine, were written in Lattice C to improve performance.

Use of Data Validity Limits. Validity limits that were established during the analysis standardization process were used to identify valid sets of data for analysis. By efficient use of validity limits, the time required to perform analysis software was greatly reduced. One example of this was the terrain-following analysis software, which used both fixed and variable limits to identify valid data sets.

Fixed validity limits involved criteria such as the maximum number of consecutive noise points allowable, the total number of noise points allowable, and other criteria specific to the given analysis task. For the terrain-following analysis software, fixed limits included a terrain-following mode of terrain-following inertial or terrain following clearance rate feedback; constant planned vertical velocity limits over the 10,000-foot segment; a limit of no more than four noisy inertial altitude values; and a limit of no more than six consecutive noisy radar values.

Variable validity limits involved criteria based on current flight conditions, such as missile speed, outside air temperature, the number of data points collected, or the distribution of data samples. For terrain-following analysis software, the total number of noisy radar altitude values was required to be less than 30 percent of the total number of data points collected for the 10,000-foot segment; also, the missile could not be on a maximum climb or dive rate for more than half of the data samples collected.

Allowing for Analyst Interaction. Several methods for allowing analyst interaction with the analysis software were considered. These included:

1. using lookup tables or interactive software inputs to select data filtering and validity limits;

2. allowing the analyst the ability to interactively overwrite "key" data values used by analysis software;

3. providing interactive plots to show the distribution of data samples before the final analysis results are calculated; and

4. allowing the analyst the ability to modify or reject the final analysis results before updating the data base or generating the flight test report.

The first two methods have not been implemented in RWT. The third method was implemented in the terrain-following analysis software by letting the analyst view an onscreen data plot; this plot shows the distribution of actual radar altitude values versus the commanded clearance altitudes and calculated terrain elevation for a given 10,000-foot segment. Figure 4 is a sample analysis plot.

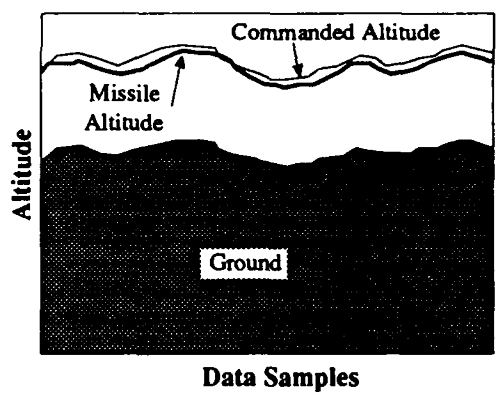

Fig. 4. Terrain-Following Analysis Plot

The fourth method for analyst interaction was also implemented in the terrain-following analysis software. Based on the distribution of data samples in the analysis plot, or the results computed by the terrain-following analysis software, the analyst has the option to accept or reject the final analysis results.

Storing Analysis Software Results

Analysis software results and critical data inputs were stored in data base tables and log files for an analyst's review and for use by report generation software. An analyst could review this information for verification of automated analysis results or for comparison with previously generated

results. Report generation software used analysis results to prepare report text, tables, and plots.

Additional Benefits of Automation

Besides reducing the time needed to perform routine flight test analysis tasks the following benefits of automated analysis were observed:

1. the results of automated analysis were more consistent between flight tests because of standardized analysis logic;

2. additional data results that would not have been possible through nonautomated techniques were generated; and

3. stored input information and analysis results improved methods of trend analysis.

For the 109D report writing tool, additional terrain-following performance results were generated as a result of the availability of online mission data and improved data filtering techniques. Additional data statistics, such as the minimum, maximum, and average radar clearance altitude over TERCOM mapsets, and the percent of noisy radar altitude samples over each accepted 10,000-foot segment, were also available through the use of automated analysis software.

Report Generation

Report generation software consists of data reduction and analysis routines combined with text, table, and plot software in a menu-driven package. The order of operations is significant because data must be reduced and results generated before tables can be built; tables must be built before text can be generated. Plot software requires the input of page numbers that may not be known until the text is generated. Report generation also includes nonautomated efforts to prepare a final, double-sided document with the proper classification markings.

Menu-driven Features

The menu-driven features of the 109D RWT guide the user through startup procedures and flight test selection, TM data reduction and filtering, and report generation.

Startup and Flight Test Selection Menus. Startup procedures consist of loading software libraries and executable functions into memory and performing table verification routines. Table verification routines are used to ensure the integrity of all primary and secondary keys in the Tomahawk data base (i.e., to ensure that all linking information between tables is valid). Statistics are also generated on the amount of data stored per flight test in each table.

The flight test selection menu allows the user to select one of the 109D flight tests to report on. A subsequent menu allows the user to review the data status for the selected flight test.

Once a flight test has been selected, the user has the option to begin TM data reduction or report generation.

TM Data Reduction Menus. TM data reduction menus guide the user through data preprocessing and data reduction subsystems. Data preprocessing consists of "working" table generation (making copies of original data tables, performing gross filtering, and data correlation), and additional filter software. Data reduction subsystems consist of automated analysis software to evaluate launch phase events, terrain-following performance, and TERCOM performance.

The software for generating data reduction menus also performs the following functions:

1. providing error messages when the required input data tables do not exist;

2. highlighting options that have not been performed yet, and providing software locks to ensure that the correct order of operations is followed; and

3. prompting the user to continue before overwriting existing data reduction and analysis results.

Report Generation Menus. Report generation menus guide the user through report section selection, and text, table, and plot generation.

The menu for text generation allows the user to edit the boilerplate text within the same RWT session. One or more devices for text output (printer, console, or disk file) are also selectable.

The menu for plot generation allows plots to be generated individually within a given report section. The plot output device (printer, console, or plotter) is also selectable. In cases of plots with multiple callouts, the callout number is chosen (e.g., terrain-following profile plot 1 of 11).

TM Data Reduction

If the TM data reduction system is selected, the user has the option to preprocess the TM data (perform filter software) or select one of the data reduction subsystem modules (perform analysis software). Both the filtering and automated analysis software provide detailed displays of the software processing status.

Filter Software. Displays for the filter software provide a processing status (current record versus the total number of records to be processed), and a status line for error messages and other important information. A count of data noise points found is maintained and the current percent of data noise is also displayed.

Example. A sample display for the inertial altitude (J950D) channel filter is shown in figure 5.

Telemetry Data Reduction System
Inertial Altitude (J950D) Channel Filter

Number of records to process :	12554
Current record being processed :	24
Number of changed values :	12
Percent Interpolated :	50.0 %

Processing...

Fig. 5. Inertial Altitude (J950D) Channel Filtering Menu

Data management. Following the successful completion of each filter module, a copy of the updated TM data table is made with a ".SAV" extension. This copy acts as a backup in case a subsequent filter module is interrupted, causing a data channel to be incompletely filtered, or if a subsequent filter module needs to be rerun with modified filter limits. In either case, the backup file is used to restore the unfiltered data values. A bell is sounded by software following the copy command.

Automated Analysis Software. Automated analysis software is run by selecting data reduction modules through the TM data reduction menu and consists of launch phase, terrain-following, and TERCOM performance analysis. Each module fills data base tables used later by report text, table, and plot software, and uses a detailed display to present the software processing status and results. Interactive analyst input is required for TERCOM performance and launch phase analysis and is optional for terrain-following analysis.

Example. A sample display for the automated terrain-following performance analysis subsystem is shown in figure 6. Information displayed in the upper section of the figure changes most rapidly as data for candidate 10,000-foot flight segments is accumulated. The lower section of the screen shows the roughness ratio, terrain category, and other results after a candidate segment is accepted.

Table Generation Software

Table generation software uses KTEXT control codes and LaserJet escape sequences to format table outputs. Occasional modifications to KTEXT control codes are needed due to problems with LaserJet Series II escape sequences. For example, extra blank lines are output when changing page orientation between portrait and landscape modes.

Fine-tuning Tables. Table generation software offers the ability to hand-edit the final report tables because

Telemetry Data Reduction System
Terrain-Following Performance
Program Status

10KSeg ddd MSN TSEG # ddd TFMODE aa Begin Set Time ddddd.ddd

TSEG Rec ddddd J941D Rec ddddd		Bad Sample Counts	
Sample Count	ddd	Noisy J950D	dd
HDOTMAX (fps)	dddd	HDOTMIN Limit	dd
HDOT (fps)	dddd	HDOTMAX Limit	dd
HDOTMIN (fps)	dddd	Max Consec. Noisy J941D	dd
Accumulated Time (Sec)	ddd.ddd	Overall Noisy J941D	dd
Accumulated Distance (Ft)	ddddd	% Overall Noisy J941D	ddd.d
Cmd Clearance HCC (Ft)	dddd	Commanded MACH	d.dddddd
Maximum Clearance (Ft)	dddd	Minimum MACH	d.dddddd
Minimum Clearance (Ft)	dddd	Radar Alt. in Specification	aa
Average Clearance (Ft)	dddd	Category aaaaaaaaa Ratio	dd.d

Searching for 10K segment ... please wait

Fig. 6. Terrain-Following Performance Reduction Program Status

tables are stored in text files prior to printing. This is useful when nonstandard data items, references, or footnotes need to be added.

Data Unavailable Tables. Table shells with a "not applicable" or "not available" data status can be generated as needed for certain flight tests; this reduces the number of changes needed to the boilerplate report format. Also, individual data status codes such as "N/AP" or "N/AV" can be displayed for individual data items within a table.

Text Generation Software

Text generation software uses KMAN commands to open all data base tables required, fill the data variables needed, produce flight-dependent text files on the hard disk, and print the boilerplate text. Boilerplate text uses evaluation commands to insert data base fields and variables computed by the text generation software, and to insert text and table files stored in other directories on the hard disk. KTEXT format commands are used to control page formats, pagination, and to reserve pages for flight test figures.

Fine-tuning Text. Fine-tuning the report text for a given flight test can be done either by modifying the boilerplate (if page formatting or boilerplate text needs to be adjusted), or by editing the final disk file output (for temporary format or text adjustments).

Plot Generation Software

Plot generation for the flight test report is a two-step process. Plot data files are produced using KMAN commands to access the flight test data tables. Lattice C programs then process the plot data files, either within RWT or at the DOS level, and use plot device drivers by GSS to output the plot on the console, printer, or plotter. Because of the memory requirements of GSS plot device drivers, data reduction and analysis software cannot be run within the same RWT session as the plot generation software. This is handled by system configuration changes, and system reboots between data reduction and plot generation sessions.

Fine-tuning Plots. Fine-tuning plots consists of fine-filtering data noise points, if any, left by the filter software. This is done either by correcting the plot data files to remove the lines containing data noise, or by marking the noisy records in the filtered TM data tables and regenerating the plot data file. The former method provides a quick, temporary correction to the data noise points while the latter provides a slower, but more permanent, correction to the problem; correcting the filtered TM data tables also ensures that other plotting or analysis software will ignore the data noise points.

Data Unavailable Plots. "Data unavailable" or "data not applicable" plots can also be generated in case of flight test data collection problems or special flight test scenarios. This reduces the number of changes needed to the boilerplate report format.

Cruise Phase Detail Plot Example. Figure 7 is a sample 109D cruise phase detail plot showing the planned and actual flight paths from launch to the first target segment, including areas of TERCOM mapsets, DSMAC scenes, and targets. The coastline and test range latitudes and longitudes are generated from canned data files -- one for each test range flown.

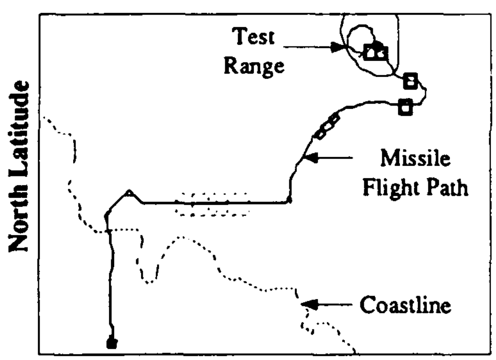

Fig. 7. Cruise Phase Detail Plot

Bomblet Impact Point/Density Analysis Example. Figure 8 is a sample 109D bomblet impact point/density analysis plot showing the distribution of bomblet impact points relative to the intended target position. The largest bomblet-free circle within a fixed area around the centroid of impact points is computed by automated analysis software and drawn by plot software. The density analysis section of the plot (not shown in the figure) is used to display the diameter of the largest bomblet-free circle, crosstrack, downtrack, and total miss distances, and the impact point pattern density.

Terrain-Following Profile Plot Example. Figure 9 is a sample 109D terrain-following profile plot showing commanded inertial and radar clearance altitude levels, actual missile-telemetered altitude levels, and the terrain profile. A bar code at the bottom of the figure indicates the commanded missile flight mode. The upper

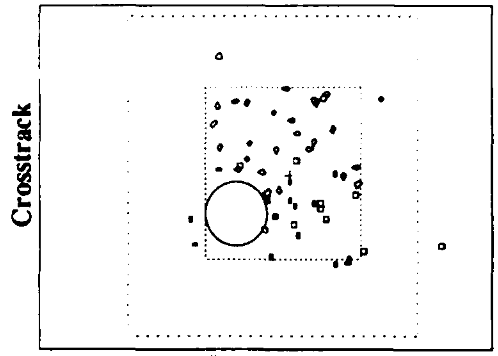

Fig. 8. Bomblet Impact Point/Density Analysis Plot

section of the figure presents a comparison of actual outside air temperature values with planned day-type temperatures at the altitudes flown. The bar code immediately below the temperature curves indicates the actual day-type conditions encountered.

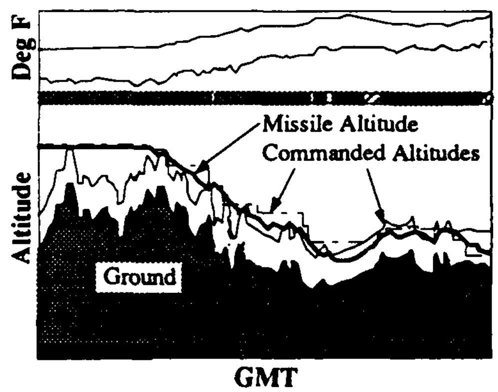

Fig. 9. Terrain-Following Profile Plot

Although the scale of the terrain-following profile plots used in the flight test report are not fine enough to support detailed engineering analysis, the plots are useful for providing a cursory evaluation of missile terrain-following performance and mission planning aggressiveness. Prior to the development of the filter techniques outlined earlier, such figures were too noisy to extract useful information for evaluation.

Nonautomated Efforts

Nonautomated report generation efforts included: reducing flight test plots to 98 percent in order to include appropriate classification markings; adding page numbers to plots when necessary; reproducing single-sided pages onto paper marked with appropriate classification labels; and making double-sided copies.

Summary

The program objectives of consistent, timely, and thorough flight test reports during the 109D DT Program

were successfully met by automating flight test analysis and report generation processes.

Existing TLAM flight test reports were reviewed and a boilerplate 109D flight test report was developed. The existing Tomahawk data base was expanded to include 109D specific data items and to support online access to TM and TSPI data. The baseline hardware and software system used for the Tomahawk data base was enhanced by adding peripheral devices such as a magnetic tape reader, LaserJet printer, and Bernoulli Box; and by adding software such as GSS graphics device drivers. Automated data entry was developed for TM and TSPI data. Data reduction techniques for TM data included automated data entry, gross filtering, GMT adjustments, and autocorrelation with planned mission data. Various heuristic techniques were used to reduce over 99 percent of the data noise points contained in individual TM channels. Existing analysis efforts were reviewed, standardized, and automated whenever possible and practical; automation was accomplished for boost-phase, TERCOM, terrain-following, and bomblet impact-point density analyses. Report generation was automated by combining data reduction and analysis routines with text, table, and plot generation software in a menu-driven package.

Additional benefits of the 109D automated flight test report writing tool include: the consistent storage of flight test range data and in-flight test results that can be used for trend analysis; improving analysis efforts by reducing tedious and time-consuming analysis functions; and making available standardized analysis procedures and documentation that can be used as a training tool for new analysts. Standardized analysis procedures and documentation can also be useful for a total quality management approach to the flight test program.

The concepts used by the 109D automated flight test report writing tool have since been applied to other variants of the Tomahawk missile with equal success.

Abbreviations and Acronyms

ADP	Automatic Data Processing
ASCII	American Standard Code for Information Interchange
CEB	Combined Effects Bomblet
CMGS	Cruise Missile Guidance Set
DOS	Disk Operating System
DSMAC	Digital Scene Matching Area Correlation
DT	Developmental Testing
DTS	Developmental Test Software
GMT	Greenwich Mean Time
GSS	Graphics Software Systems
I/O	Input/Output
J984U	Telemetry Commanded Clearance Pointer Measurement Number
KMAN	KnowledgeMan
MB	Megabyte
NUSC	Naval Underwater Systems Center
RAM	Random Access Memory
RWT	Report Writing Tool
SPS	Samples per Second
TERCOM	Terrain Contour Matching
TLAM	Tomahawk Land-Attack Missile
TM	Telemetry
TSPI	Time, Space, and Position Information

References

1. OPNAV Instruction 5239.1A: Department of the Navy Automatic Data Processing Security Program. Washington, DC: Chief of Naval Operations, 1 April 1985

2. KnowledgeMan Technical Reference: The Universal Knowledge Management System Version 2.5. Lafayette, Indiana: **mdbs**, Inc., 1988.

3. R. S. Burington and D. C. May. *Handbook of Probability and Statistics with Tables*. Sandusky, Ohio: Handbook Publishers, 1953.

4. Tomahawk Report Writing Tool Analyst and User Guide Version 2.00. Newport, Rhode Island: NUSC, 1989.

5. J. D. Kraus. *Electromagnetics*. 3d ed. McGraw-Hill, 1984.

6. S. E. Goodman and S. T. Hedetniemi. *Introduction to the Design and Analysis of Algorithms*. McGraw-Hill, 1977.